

[← Previous](#) [Feedback](#) [Share](#)[Next →](#)

# Populate Dropdown with Products

Beginner 10 min.

[SAP Build, Beginner, Tutorial,](#)

[SAP Business Technology Platform, SAP Build Apps](#)

Populate a dropdown field with product data, so that you can provide the user with a choice field.

You will learn

- ✓ How to create a data source to SAP data
- ✓ How to make a call to an SAP backend API
- ✓ How to populate a dropdown list from backend data



**Daniel Wroblewski**

April 22, 2025

**Created by**



March 1, 2023

**Contributors**



Step 1: Create a destination to the CAP service

**Step 2: Create a data resource**

Step 3: Create data variable

Step 4: Create dropdown for materials

Step 5: Run app

[Back to Top](#)

## Prerequisites

- You have completed the the tutorial [Create SAP Build App to Trigger Workflow](#)

Previously, we created an input box for the user to enter a material ID for the sales order. Now, we want to make a call to an SAP backend, retrieve product data, and populate a dropdown box so the user can choose from a list of products.

[Open all](#)    [Close all](#)

### STEP 1

#### Create a destination to the CAP service

You will use a CAP service that we created for this tutorial, which contains an entity called **Products**, to population a dropdown box.

Follow the instructions in Step 4 of [Set Up Prerequisites for SAP Build CodeJam](#).

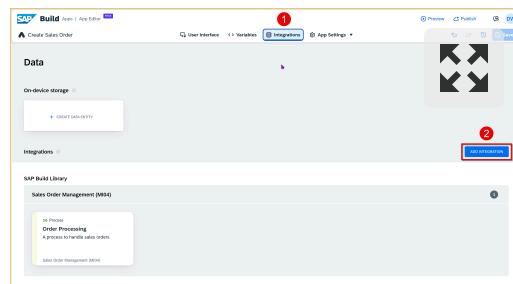
Completed

✓ STEP 2

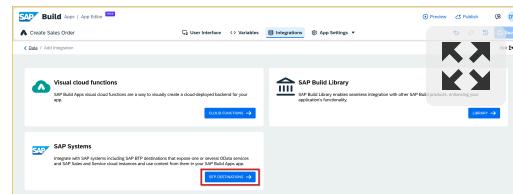
## Create a data resource

A data resource defines a connection to a backend, and when executed, will bring back the data we want. Here, we need to define the connection to the CAP service that contains our products.

1. Go to the **Integrations** tab, and the **Add Integration**.

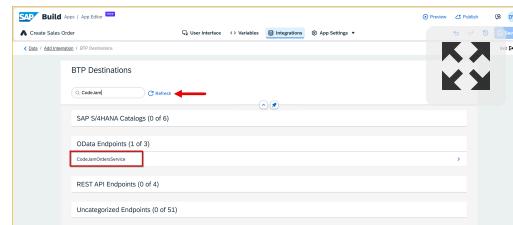


2. Click **BTP Destinations**.

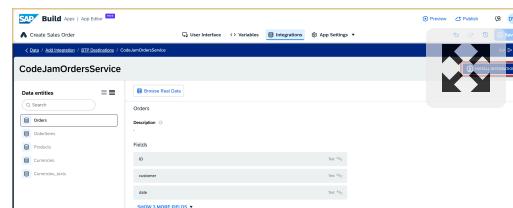


3. Select your **CodeJamOrdersService** destination.

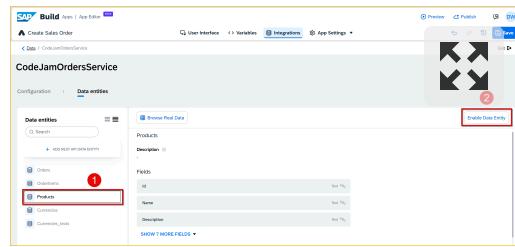
If you just created the destination, you may need to click **Refresh**.



4. Click **Install Integration**.



Select the **Products** entity on the left, and click **Enable Data Entity** on the right.



5. Click **Save** (upper right).

Completed

### STEP 3

## Create data variable

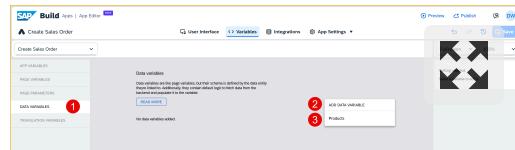
After retrieving the data, you need a place to put it, with the proper schema so we can reference all its fields.

1. Click **Variables**.



2. On the left, click **Data Variables**.

Then click **Add Data Variable**, and select **Products**.



You should get a new data variable defined for your **Products** entity.



3. Click **Save** (upper right).

Completed

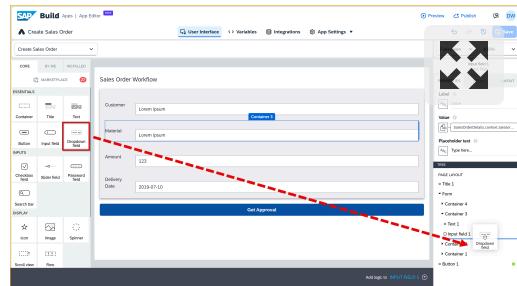


STEP 4

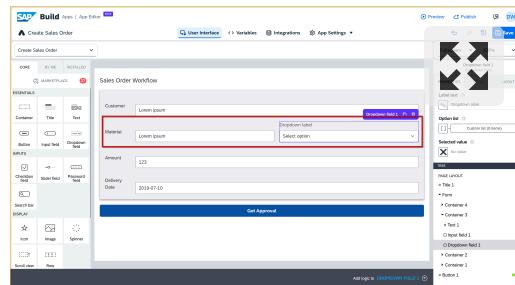
# Create dropdown for materials

1. Click **User Interface** to return to the UI of the page.
2. Drag a dropdown field next to the input box for the **Material**.

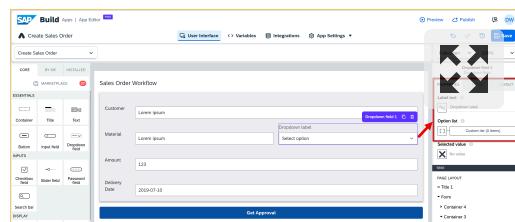
You will likely have to select the **Material** input field and then expand the tree view on the bottom right. Also make sure for **Container 3**, for the material, you have text, input, and dropdown components at the same level.



It should look like this.



3. Click on the dropdown box, and change the following properties.



Field	Value
<b>Label text</b>	Delete <b>Dropdown label</b> .
<b>Option list</b>	Click the list of values icon, and set the binding to a formula, and set the formula to the following:

```
MAP(data.Products1,
{label: item.Name,
value: item.Id})
```

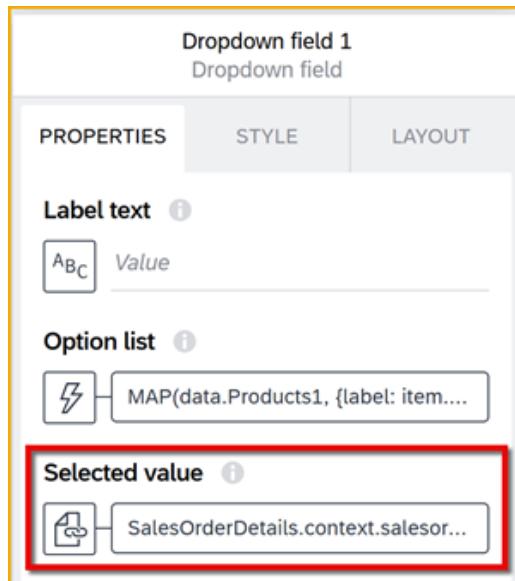
Field	Value
	 <p>Click <b>Save</b> twice on the formula pop up.</p>

### What does the formula do?

A dropdown field wants the options as a list of objects, with each object having 2 fields: `Name` for the display name, and `Id` for the underlying value of the option. The `Map` function takes the data in the data variable and converts it to the format we need.

4. You will have to bind the dropdown field to the `SalesOrderDetails` page variable (as we did for the input field for material).

From the properties tab, for the **Selected value** field, set the binding to: **Data and variables > Page variable > SalesOrderDetails > context > salesorderdetails > material.**



Click **Save**.

5. Click **Save** (upper right).

Completed

 STEP 5[Run app](#)

## Next Steps

 TUTORIAL[Check Status of Process  
from SAP Build App](#) 20 min.[Developer Products](#)[Trials & Downloads](#)[Site Information](#)[ABAP Platform](#)[ABAP Development  
Tools](#)[Privacy](#)

SAP Business  
Application Studio

Mobile Development  
Kit Client

Terms of Use

Legal Disclosure

SAP Business  
Technology Platform

SAP Business  
Application Studio

Copyright

SAP Conversational AI

SAP Data Intelligence  
Trial

Trademark

SAP Data Intelligence

Newsletter

CONTINUE

PREVIOUS

© 2025 SAP SE or an SAP affiliate company. All rights reserved.

